FCBBooster

user manual

version 2.0.0

© Copyright 2010-2013 **Amedeo Farello**. All rights reserved.

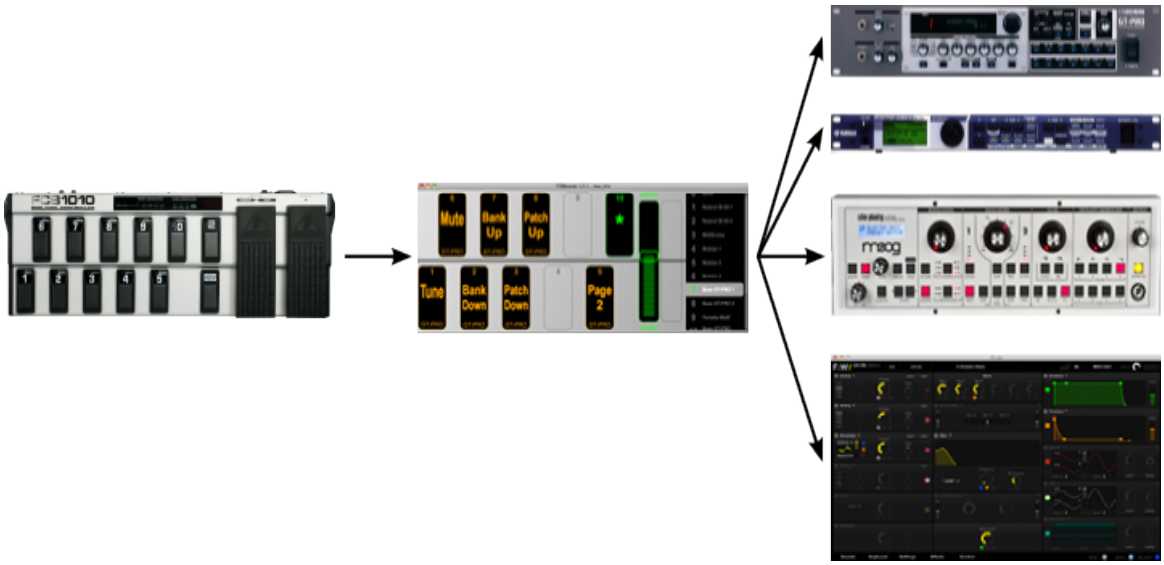
Table of contents

Introduction....3
Features....3
Getting started....4
Changes since version 1.2.1....6
Sandboxing....6
New features....6
Concepts....8
Targets....8
Actions, states, controls and patches....8
Reference bank....9
Navigation mode....10
Default document....11
Interfacing with the FCB 1010....12
Input....12
Automatic learning....12
Step-by-step learning....13
The main window....15
Programming....16
The editor window....16
The "patches" section....16
The "controls" section....17
The "states" section....18
The "actions" section....19
Common properties for footswitches and expression pedals....19
Actions for footswitches....20
midi program change....20
midi control change....20
midi note....20
midi sysex....20
midi all sound off....20
midi all notes off....20
midi reset controllers....20
go to patch....21
mute expression pedal....21
wait....21
Actions for expression pedals....21
midi control change....21
midi sysex....21
midi channel pressure....21
MIDI Implementation Reference....23

Introduction

Features

FCBBooster is a software tool conceived to increase the performance of the Behringer FCB 1010 MIDI foot controller. It works as an interface between the pedalboard and whatever MIDI device, hardware or software, you want to drive with it, like synthesizers, drum machines, effects, loopers, etc. If your setup includes a Mac and an FCB 1010, this application will greatly augment the pedalboard's capabilities and simplify its use at once without any hardware modification.



FCBBooster is not a software editor for the Behringer pedalboard, altough it is able to receive, send, save and load FCB 1010 SysEx dumps. Actually, using FCBBooster means that you won't need to directly program the FCB 1010 pedalboard anymore, either through an editor or a sequence of footswitch presses, the so called “river dance”.

Speaking about MIDI messages, the following table compares the FCB 1010 and FCBBooster capabilities:

		FCB 1010	**FCBBooster**
memory		10 banks	32768 patches
number of messages that a footswitch can send simultaneously	program change	5	unlimited
	control change	2	unlimited
	note	1	unlimited
	system exclusive	not available	unlimited
	all sound off	not available	unlimited
	all notes off	not available	unlimited
	reset all controllers	not available	unlimited
number of messages that an expression pedal can send simultaneously	control change	1	unlimited
	system exclusive	not available	unlimited
	channel pressure	not available	unlimited

FCBBooster's main features are:

- ability to send a virtually unlimited number of MIDI messages to a virtually unlimited number of MIDI devices simultaneously from each footswitch or expression pedal
- ability to send MIDI messages that the FCB 1010 is not able to manage, including system exclusive messages, both fixed and variable
- simple but powerful programming capabilities, implemented through a WYSIWYG editor, allow you to define complex functions that would be impossible to handle for a FCB 1010
- gives a visual feedback of the functions assigned to footswitches and expression pedals
- manages FCB 1010 SysEx dumps for an easy interfacing with the pedalboard and backup purposes
- the order and timing of MIDI messages can be carefully programmed
- footswitches can act as buttons, switches or multiple state switches
- messages can be individually programmed to be sent when a footswitch is either pressed or released
- much easier and faster to program than the actual FCB 1010
- entire setups can be saved as documents, allowing backups and fast switching between completely different configurations

FCBBooster is a 32/64-bit Intel fat binary that runs on Mac OS X 10.6.6 (Snow Leopard) and later versions. It requires a minimum screen resolution of 1280 x 800.

FCBBooster should be able to support pedalboards modified with the UnO chip, as long as it is possible to program them as described in the "Interfacing with the FCB 1010" chapter. Of course you will need to use the "step-by-step" learning approach.

Getting started

You should read the "Concept" chapter to familiarize with FCBBooster's way of doing things, then the "Reference bank" and "Input" sections of the "Interfacing with the FCB 1010" chapter, to learn how to connect the application with your pedalboard.

When you have succesfully connected FCBBooster and the FCB 1010, the quickest way to send your first MIDI message is:

1. open the preferences window with the proper command in the "FCBBooster" menu and select the "targets" panel
2. click the "new" button to create a new target
3. open the target window with the "edit" button or by double clicking the target row in the list
4. choose the preferred MIDI device between those available and select a MIDI transmission channel, then press the "OK" button
5. close the preferences window
6. open the editor window with the proper command in the "Edit" menu or by double clicking

the “0 Untitled” patch in the list at the right side of the main window

7. in the “controls” section of the editor window, click the “new” button and choose “footswitch 1” to create a new footswitch and its default state
8. in the “actions” section of the editor window, click the “new” button and choose “midi note” to add a new MIDI note message to the default state of footswitch #1
9. change the “number” property to “60 – C3” to emit a middle C; the target property should indicate the target you just created in the preferences window
10. close the editor window
11. press footswitch #1 on the FCB 1010: the C3 note should be received by the target until you hold down the footswitch, while the corresponding button inside the FCBBooster window is highlighted

FCBBooster has a few sample .fcb files embedded in it. To save them on your computer, please use the “Extract Examples...” command of the “File” menu.

MIDI is a trademark of MIDI Manufacturers Association. Behringer is a trademark of MUSIC Group IP Ltd.

Changes since version 1.2.1

Sandboxing

As prescribed by the Mac App Store rules, FCBBooster is now a "sandboxed" application. Discussing the OS X sandbox in detail is beyond the scope of this manual, but I just want to point out that to make the application compatible with the sandbox AND maintain all its functions unaltered would have meant losing compatibility with OS X 10.6 (Snow Leopard), something I was not willing to do.

So these are the consequences of "sandboxing":

- the preferences file stored by FCBBooster versions up to and including 1.2.1 will be moved to its new location inside the application sandbox the first time you start the application; if, for any reason, you want to keep your old preferences file, you must duplicate it and store it away from the (now obsolete) application preferences folder in Library/Application Support before you launch the new version;
- the "default document" is not just a reference to some user specified document anymore; instead, it is a concrete document residing in the application sandbox; to keep using the default document from previous versions, just open it normally, then choose the "Save As Default" command of the "File" menu;
- the "Open Recent" menu remembers documents opened during the current session, but not those that were opened in previous sessions;
- since the standard version and the demo version use different sandboxes, they do not share preferences anymore.

New features

- MIDI destination devices and channels are now coupled as "targets" and defined just once in the preferences window, instead of being specified multiple times on an action-by-action basis;

  actions just choose their target from a menu and thus can be much more easily adapted to a change in the MIDI configuration of your system;

  the now obsolete "Edit/Find MIDI Devices..." command has therefore been removed, while the "label" and "target" properties of controls have been renamed as "primary label" and "secondary label" to avoid ambiguities;

  when a file written with an earlier version is loaded, the necessary targets are created according to the properties of the actions found
- multiple state footswitches can now optionally expose their current state to all actions belonging to the same patch, so that these can be programmed to perform only when a control is in a particular state
- interfacing with the FCB 1010 has become much easier: FCBBooster is now able to receive a SysEx dump from the pedalboard or load it from a file, apply the possible needed corrections and send it back without requiring any programming on the FCB 1010;

  the old step-by-step learn mode is anyway still available;

FCBBooster can also save SysEx dumps to a file for backup purposes

- the "patches setup" window, now simply called the editor window, has a new, clearer layout, with green lines and boxes indicating the relationships between patches, controls, states and actions;

  when the editor window is open, the currently selected control is shown with a red border in the main window;

  there are new "cut", "copy" and "paste" buttons for each of the "patches", "controls", "state" and "actions" sections; the "controls" section also has a new "swap" button to easily exchange position between controls of the same kind

- a new "reference bank" preference allows to choose between the traditional mode of operation, that uses a single FCB 1010 bank as source for MIDI messages, and a new "all banks" mode, that limits the number of the definable patches to ten, but lets you select them with the up/down keys on the pedalboard

- some new MIDI actions are available: "all sound off", "all notes off", "reset all controllers" for footswitches and "channel pressure" for expression pedals

- a new "wait" action may be used to introduce a delay between actions; this can be useful to let some hardware devices digest complex sysex messages before sending them other messages

- there is a new, unified preferences window

- the "navigation" mode is now not immediately dismissed when a footswitch is pressed; instead it stays active until the "*" activation footswitch is pressed again; this makes this mode much more effective

  the current destination patch is highlighted with a green border in the list at the right side of the main window

- the "go to patch" action now supports special destinations like "higher in level", "last visited", "next visited", "previous in list", "next in list", which are analogous to the navigation mode commands

- when the footswitch that activates the navigation mode is changed in the preferences window, any definition using that location is now always moved to a free position, if it is available

- the .fcb file format has changed to support the new features

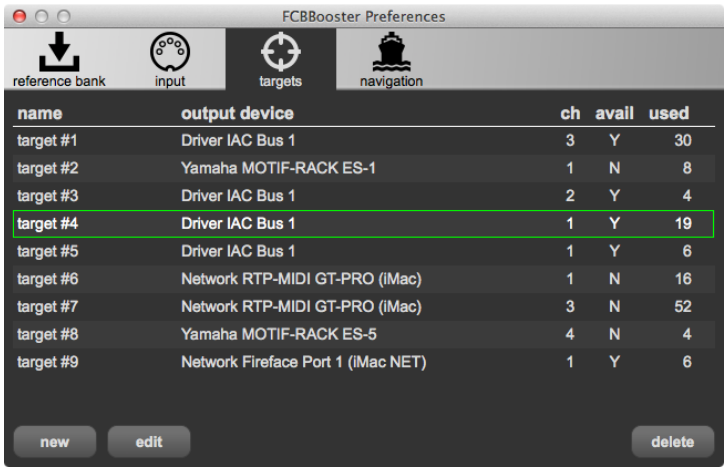
Concepts

This section explains some basic concepts you need to be aware of when using FCBBooster. In this manual the terms "footswitch", "expression pedal" and "up/down key" follow the terminology of the Behringer FCB 1010 manual.

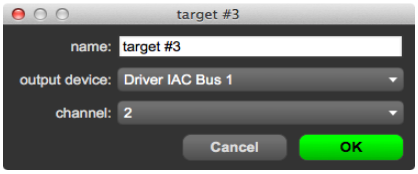
Targets

FCBBooster's task is to send MIDI messages to MIDI devices. A target brings together, under a user specified name, the name of a MIDI device with a MIDI transmission channel. The target's name can be subsequently used by MIDI actions (more on these later) to define their destinations.

Targets are created and edited in the "targets" panel of the preferences window, that shows the list of the defined targets.


FCBBooster Preferences
reference bank
input
targets
navigation
name
output device
ch
avail
used
target #1 Driver IAC Bus 1 3 Y 30
target #2 Yamaha MOTIF-RACK ES-1 1 N 8
target #3 Driver IAC Bus 1 2 Y 4
target #4 Driver IAC Bus 1 1 Y 19
target #5 Driver IAC Bus 1 1 Y 6
target #6 Network RTP-MIDI GT-PRO (iMac) 1 N 16
target #7 Network RTP-MIDI GT-PRO (iMac) 3 N 52
target #8 Yamaha MOTIF-RACK ES-5 4 N 4
target #9 Network Fireface Port 1 (iMac NET) 1 Y 6
new
edit
delete

In addition to the name, device and channel of each target, the list indicates whether the related device is available and the number of actions that currently use the target. Pressing the "edit" button or double clicking an item opens a window to change its settings.



Each target has a unique name.

Actions, states, controls and patches

The operation of FCBBooster revolves around a hierarchy of four types of elements: "patches", "controls", "states" and "actions".

Actions are at the lowest level and usually describe a single MIDI message sent to a specific target.

Actions are contained in a **state**. A single state can contain from zero to a virtually unlimited number of actions.

States are contained by **controls** (footswitches and expression pedals), which directly represent their physical counterparts on the pedalboard. While an expression pedal always has exactly one

state, a footswitch can contain a number of states variable between 1 and 256.

A footswitch with a single state behaves like a button, a two-state footswitch is equivalent to an on/off switch, while a footswitch with three or more states is a selector that, each time that it is pressed, passes to the next state and at the end of the cycle starts again from state #1.

When a control has more than one state, its current state can be optionally made visibile to actions. In this way it is possible to choose whether a particular action should be performed only when a control is in a particular state.

A control can also be defined as a simple alias for another control. This not only makes it easy to insert the same control in different places, but with multi-state controls and the ability to perform actions conditionally, allows the programming of complex operations with ease.

Controls are contained in **patches**. Since the FCB 1010 has 10 footswitches and 2 expression pedals, a patch can contain at most that number of controls. When a control is defined for the current patch, it appears inside FCBBooster's main window according to the properties it was given in the editor window. Undefined controls are represented with just an outline.

An FCBBooster document contains a number of patches between 1 and 32768. There is always a "current" patch, which is shown in the main window.

The current patch completely describes the functions assigned to the controls of the FCB 1010 at the moment. Since each state can contain multiple actions, pressing a footswitch or changing the position of an expression pedal can generate a complex sequence of MIDI messages. Depending on the way you programmed your patches, selecting another patch and activating the same control could generate a completely different sequence.

Depending on the "reference bank" setting, the current patch can also be selected using the "navigation" features or the pedalboard up/down keys. Last but non least, a patch can be selected executing a "go to patch" action.

See the "Programming" chapter to learn how to properly setup these elements.

Reference bank

There are two ways to use the application in relation to the FCB 1010 banks: using a single reference bank or all the ten banks. Each way has its advantages and disadvantages.

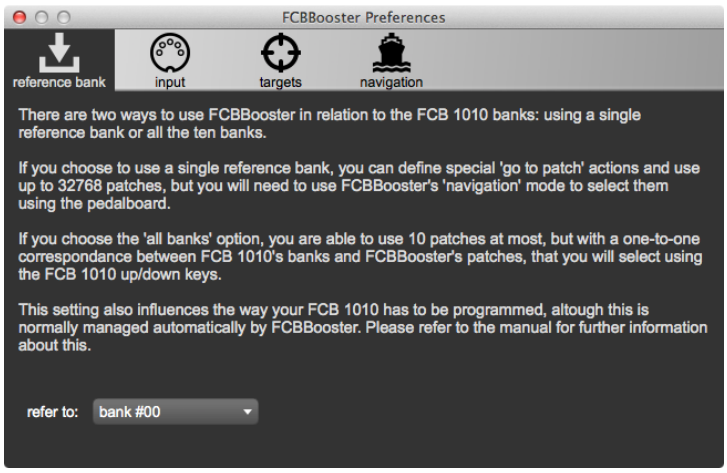
	using a single bank	using all banks
number of FCB 1010 banks that need to be programmed in a way compatible with FCBBooster	1, at user's choice	all 10 banks
maximum number of selectable FCBBooster patches	all (up to 32768)	10
navigation mode available	yes	no
"go to patch" actions available	yes	no
patches selectable with mouse	yes	no
patches selectable with FCB 1010 up/down keys	no	yes

In other words, if you choose to use a single reference bank, you can make the most of the application and use the remaining 9 banks on the pedalboard the way you want, but selecting patches with your feet is a bit less straightforward, since you need to use the "navigation" mode.

If you choose to use all banks, you loose some power, but you have a one-to-one correspondance

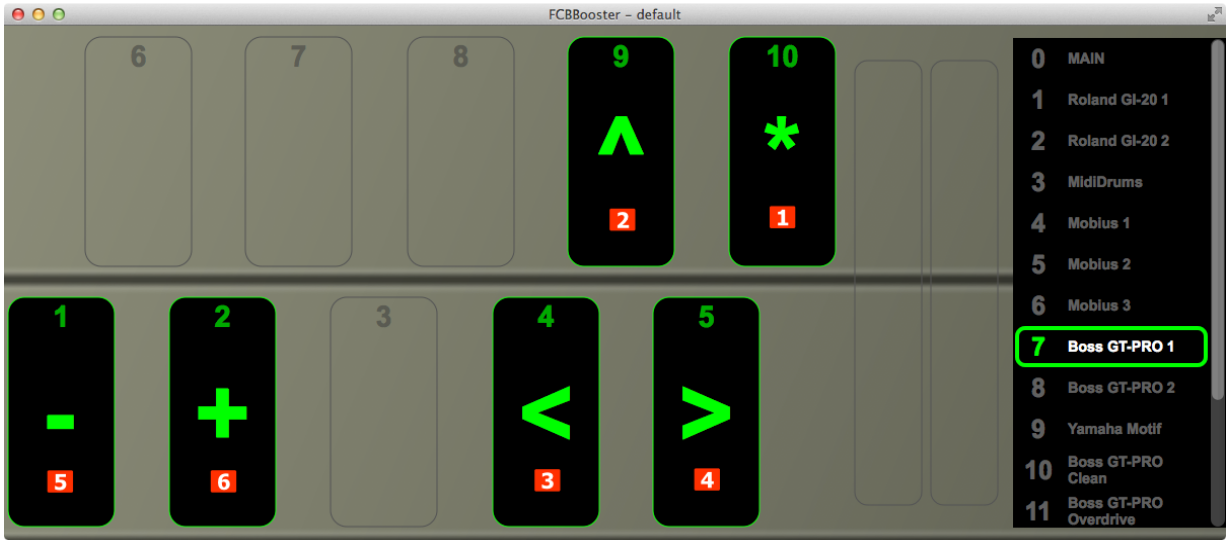
between FCBBooster patches and FCB 1010 banks, which is handy.

You can choose this setting in the "reference bank" panel of the preferences window.


FCBBooster Preferences
reference bank
input
targets
navigation
There are two ways to use FCBBooster in relation to the FCB 1010 banks: using a single reference bank or all the ten banks.
If you choose to use a single reference bank, you can define special 'go to patch' actions and use up to 32768 patches, but you will need to use FCBBooster's 'navigation' mode to select them using the pedalboard.
If you choose the 'all banks' option, you are able to use 10 patches at most, but with a one-to-one correspondance between FCB 1010's banks and FCBBooster's patches, that you will select using the FCB 1010 up/down keys.
This setting also influences the way your FCB 1010 has to be programmed, altough this is normally managed automatically by FCBBooster. Please refer to the manual for further information about this.
refer to:
bank #00

Navigation mode

This mode is useful to select a patch without resorting to the mouse. It is not available when the "reference bank" preference is set to "all banks", since that means that is the FCB 1010 to set the current patch, which coincides with the current pedalboard's bank.



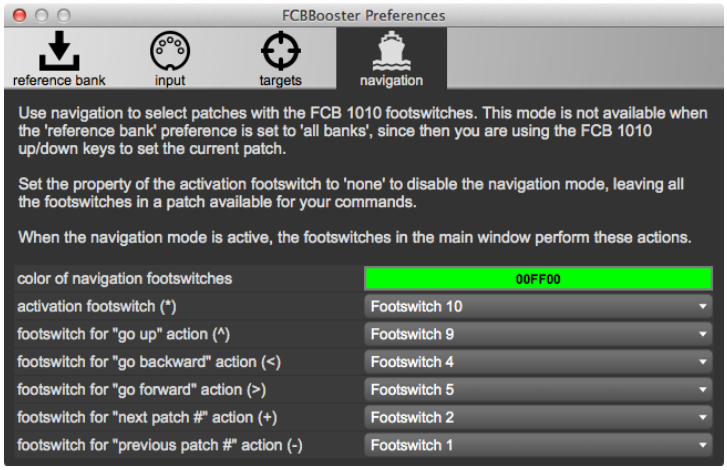
When this mode is active, the current patch in the main FCBBooster's window is substituted with a special one, whose footswitches execute the actions described in the following table.

footswitch	action
[1] on/off (*)	this footswitch is used to activate or deactivate the navigation mode, it will appear in all patches, superimposed on any existing control definition; set its value to "none" to hide it and disable the navigation mode, leaving all the footswitches in a patch available for your commands
[2] go up (^)	this footswitch is used to select a patch with a higher rank than the current patch, if any; a higher rank patch is a patch that contains at least one "go to patch" action pointing to the current patch and has a lower level number; '0' represents the highest rank;

	set its value to "none" to hide the footswitch in the navigation patch
[3] go backward (<)	this footswitch is used to select the previous patch in the list of the visited patches; set its value to "none" to hide the footswitch in the navigation patch
[4] go forward (>)	this footswitch is used to select the next patch in the list of the visited patches; set its value to "none" to hide the footswitch in the navigation patch
[5] previous patch (-)	this footswitch is used to select the next patch in the list of the existing patches; set its value to "none" to hide the footswitch in the navigation patch
[6] next patch (+)	this footswitch is used to select the previous patch in the list of the existing patches; set its value to "none" to hide the footswitch in the navigation patch

While in navigation mode, use footswitches #2 to #6 to select the destination patch, which is highlighted with a green border in the list at the right side of the main window. Use footswitch #1 to confirm your choice and exit navigation mode.

You can choose which footswitch executes a particular navigation action and the color of the navigation footswitches using the "navigation" panel of the preferences window.


FCBBooster Preferences
reference bank
input
targets
navigation
Use navigation to select patches with the FCB 1010 footswitches. This mode is not available when the 'reference bank' preference is set to 'all banks', since then you are using the FCB 1010 up/down keys to set the current patch.
Set the property of the activation footswitch to 'none' to disable the navigation mode, leaving all the footswitches in a patch available for your commands.
When the navigation mode is active, the footswitches in the main window perform these actions.
color of navigation footswitches
00FF00
activation footswitch (*)
Footswitch 10
footswitch for "go up" action (^)
Footswitch 9
footswitch for "go backward" action (<)
Footswitch 4
footswitch for "go forward" action (>)
Footswitch 5
footswitch for "next patch #" action (+)
Footswitch 2
footswitch for "previous patch #" action (-)
Footswitch 1

Default document

If you keep using the same .fcb document each time you use FCBBooster, save that file as the "default document", using the "Save As Default" command of the "File" menu. From this moment on, whenever you start the FCBBooster application, it will load that document instead of just creating a new, empty one. To reset the default document to an empty one, you can obviously just choose "New" and then "Save As Default". Please note that DEMO versions don't load the default document at startup.

Interfacing with the FCB 1010

Input

FCBBooster intercepts MIDI notes and control change messages sent by the FCB 1010 footswitches and expression pedals and uses them to activate the programmed sequence of "actions" for the corresponding control in the current patch.

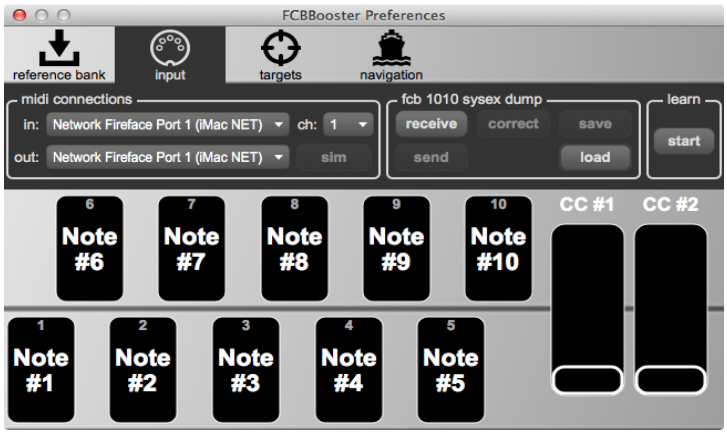
For this to work, each FCB 1010 footswitch must be programmed to send a unique MIDI note, each pedal must send a constant and unique control change message and the application has to know which control sends what.

The following table shows a possible FCB 1010 setup compatible with FCBBooster. Please note that the Program Change and Control Change settings of footswitches are irrelevant and thus are not shown. Only a single bank is displayed, note that if you choose "all banks" for the "reference bank" preference, all ten banks must be programmed equally.

Preset #	Expression pedal A			Expression pedal B			Note
	CC	min	max	CC	min	max	
1	1	0	127	2	0	127	1
2	1	0	127	2	0	127	2
3	1	0	127	2	0	127	3
4	1	0	127	2	0	127	4
5	1	0	127	2	0	127	5
6	1	0	127	2	0	127	6
7	1	0	127	2	0	127	7
8	1	0	127	2	0	127	8
9	1	0	127	2	0	127	9
10	1	0	127	2	0	127	10

Note also that the MIDI transmission channel for all messages must be the same.

The FCB 1010 - FCBBooster connection is established through the "input" panel of the preferences window. The operation needs to be executed only once, unless you change something in your FCB 1010's programming, or your MIDI connections, or you delete FCBBooster's preferences.



Automatic learning

The preferred way to connect FCBBooster to your FCB 1010 requires that both an input and an

output MIDI connection to the pedalboard is available.

Please note that you should be aware of the current “reference bank” preference before starting, since this influences the way your pedalboard must be set up.

1. connect and power the FCB 1010
2. open the FCBBooster preferences window and select the “input” panel
3. in the “midi connections” box, select the appropriate MIDI input and output device
4. press the “receive” button in the “fcb 1010 sysex dump” box
5. an alert window will appear, informing that you are entering the learn mode: press the “OK” button, your pedalboard will be requested to send a SysEx dump containing its current settings; this will produce one of the following results:

a) “dump successfully received”: FCBBooster is ready to work with your pedalboard.

b) “Incompatible settings”: a dump was succesfully received, but the current settings of your FCB 1010 aren't compatible with FCBBooster's needs;

   to solve the problem, press the “correct” button, then use the “send” button to upload the new settings into your FCB 1010 and follow the instructions.

   ATTENTION: be aware that you will loose some of the current pedalboard settings, so you may want to make a backup before proceeding.

c) “Time out”: FCBBooster was unable to receive a dump from your FCB 1010; this is probably due to incorrect MIDI connections or to a defective MIDI interface;

   if you cannot solve the problem you will need to program the FCB 1010 yourself according to the FCBBooster's needs and then resort to step-by-step learning.

You can use the “load” and “save” buttons in the “fcb 1010 sysex dump” box to load an FCB 1010 SysEx dump from a file or to archive the current dump. Note that loading a SysEx file is equivalent to receive a SysEx dump through a MIDI connection.

Please note that any SysEx dump received via a MIDI connection or loaded from a file will be lost if you select another preferences panel or exit the preferences window. Information concerning the messages sent from your pedalboard are obviously retained.

Step-by-step learning

There is an alternative way to teach FCBBooster about what notes and control change messages to expect from your FCB 1010. There is no need to use it, unless for some reason the preferred way failed.

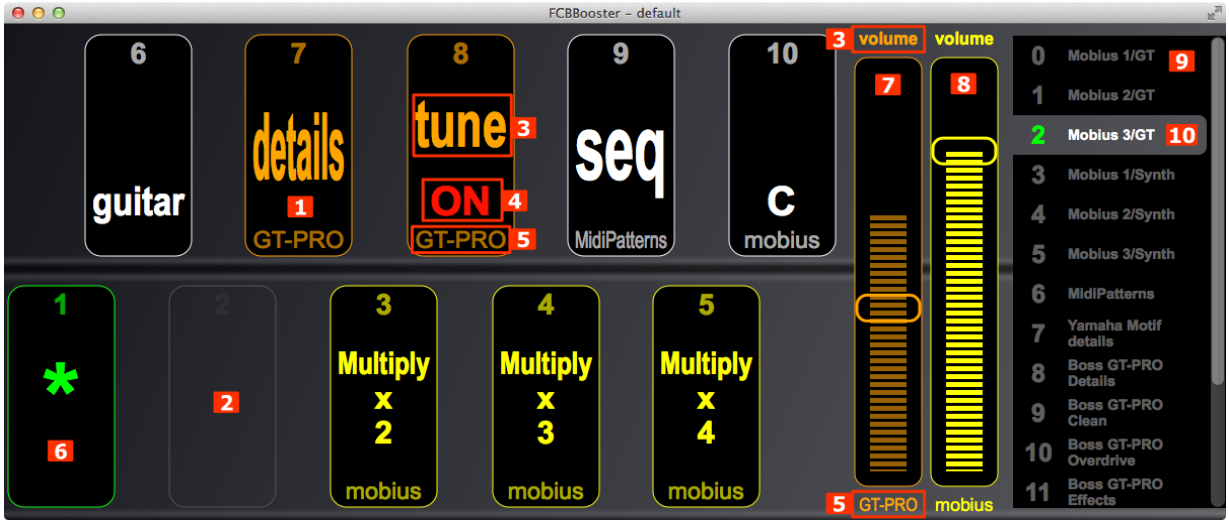
1. connect and power the FCB 1010
2. open the FCBBooster preferences window and select the “input” panel
3. in the “midi connections” box, select the appropriate MIDI input device and channel
4. press the “start” button in the “learn” box
5. an alert window will appear, informing that you are entering the learn mode: press the “OK” button, the gray “start” button will turn into a green “stop” button
6. one after the other, pedals and footswitches in the preferences window will light up in

green: just press or move the corresponding control on the FCB 1010 to connect it to FCBBooster

7. FCBBooster will automatically exit the learn mode after receiving a message from the last pedal in the sequence; you can abort the procedure anytime by pressing the "stop" button

The main window

This resizable window is made of two parts: the left side mimics the FCB 1010 layout and shows the currently selected patch, giving a visual feedback of the state of the controls, while the right side shows a scrollable list of all the existing patches, in which the current patch is highlighted.



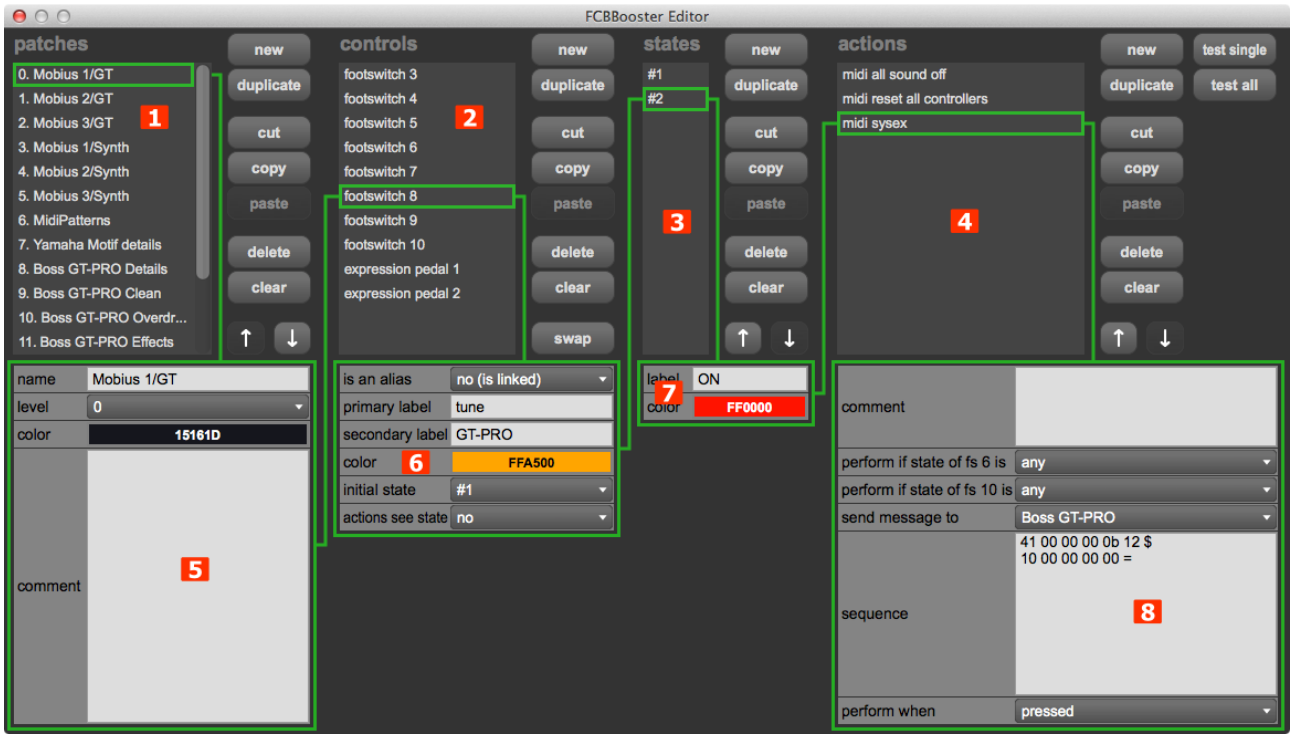
The sample window above shows various aspects:

1. This is a single state footswitch. Its color is user assignable. When it gets clicked or the real footswitch on the pedalboard is pressed, its border is highlighted.
2. This is an undefined footswitch. Pressing the corresponding footswitch on the pedalboard has no effect.
3. This is the text assigned to the "primary label" property of the control.
4. This is the name of the current state of a multiple state footswitch. Its color is user assignable.
5. This is the text assigned to the "secondary label" property of the control.
6. This footswitch activates the navigation mode. Presence, color and position of this switch are at the user's discretion and are set inside the "navigation" panel of the preferences window.
7. This is an expression pedal. Its color is user assignable. The segmented bar indicating its current position is dimmed because the actual pedal it refers to has been moved while another patch was selected. Acting on the pedal won't have any effect until its position again matches the position it had the last time it was used with the current patch.
8. This is an another expression pedal. The segmented bar indicating its current position is drawn normally because its current value corresponds to that of the actual pedal it refers to on the FCB 1010.
9. This is the list of all the patches that are defined in the current document. Click on a name to select the corresponding patch or double click it to open the editor.
10. This is the current patch, the one whose properties are show in the left part of the window. Each patch may have a different background color.

Programming

The editor window

All programming takes place inside the editor window. The window contains four sections respectively dedicated to patches, controls, states and actions. Please refer to the "Concepts" chaper to learn more about these elements and the hierarchy that links them.



Each section contains a list of selectable items and a few buttons in the upper part and a list of properties, belonging to the selected item in the list, at the bottom. A sequence of green boxes and connection lines shows the relationships between the items selected at the moment.

1. This is the list of the defined patches.
2. This is the list of the defined controls for the currently selected patch.
3. This is the list of the defined states for the currently selected control.
4. This is the list of the defined actions for the currently selected state.
5. These are the properties of the currently selected patch.
6. These are the properties of the currently selected control.
7. These are the properties of the currently selected state.
8. These are the properties of the currently selected action.

Each section has its own individual clipboard.

The "patches" section

A patch defines the behaviour of the 10 footswitches and the 2 expression pedals of a FCB 1010, collectively called "controls".

Each patch is identified by a unique progressive number, which is automatically assigned, and by a user assignable name. The name does not need to be unique, altough this is surely advisable.

This section contains the following buttons.

new	creates a new, empty patch
duplicate	duplicates the currently selected patch
cut	copies the currently selected patch into the clipboard and deletes it from the list
copy	copies the currently selected patch into the clipboard
paste	creates a new patch as a copy of that contained in the clipboard
delete	deletes the currently selected patch
clear	deletes all the patches and creates a new, empty patch
arrows	move the currently selected patch up or down in the list

Each patch has the following user assignable properties.

name	text that identifies the patch in the main window list
level	this value determines the operation of the "go up" (^) footswitch in the navigation mode (see the command 'Setup/Navigation...'); when in navigation mode, the (^) footswitch will select a patch with a higher rank than the current patch, if any; a higher rank patch is a patch that contains at least one "go to patch" action pointing to the current patch and has a lower level number; '0' represents the highest rank
color	the color used to paint the background of the patch; there is a library of specific colors for the backgrounds of the patches; using the menu that appears when you click on the color, you can choose a color from those available in the library, or add a new one, or modify or delete existing colors
comment	you can use this space for notes about the patch

The "controls" section

Controls in FCBBooster correspond to those of the FCB 1010: footswitches and expression pedals. The appearance of controls is customizable to enhance their visibility.

In addition to their normal operation, controls can be programmed to act as an "alias" for another control. This can be useful to insert the same control inside different patches, ensuring that each instance remains synchronized with the other.

This section contains the following buttons.

new	creates a new, empty control; clicking on the button opens a menu to choose which control to create between those not already existing
duplicate	duplicates the currently selected control; clicking on the button opens a menu to choose which control to create between those not already existing
cut	copies the currently selected control into the clipboard and deletes it from the list
copy	copies the currently selected control into the clipboard

paste	creates a new control as a copy of that contained in the clipboard; clicking on the button opens a menu to choose which control to create between those not already existing
delete	deletes the currently selected control
clear	deletes all the controls
swap	switches the positions of the selected control with that of another in the list; clicking on the button opens a menu to choose the other control

These are the properties that can be assigned to *all* controls:

is an alias	when this property is 'yes', the control is simply a 'link' to another control defined elsewhere, usually inside a different patch; in this case both its appearance and its functions will be inherited from the reference control and cannot be modified locally

These are the properties that can be assigned to a control which is not an *alias*:

primary label	the primary text that identifies the control in the main window
secondary label	a secondary text to identify the control; could be used, for example, to indicate the recipient of the actions defined for the control
color	the color used to paint the control; there is a library of specific colors for the controls: using the menu that appears when you click on the color, you can choose a color from those available in the library, or add a new one, or modify or delete existing colors
initial state (footswitches only)	this property is only available when the footswitch has more than one state and indicates the state that is assigned to the footswitch when the document is loaded
actions see states (footswitches only)	this property is only available when the footswitch has more than one state and indicates if the actions belonging to the same patch, but not to this control, will be able to check which state of this control is the current one and behave accordingly
initial value (expr. pedals only)	indicates the initial value that is assigned to the expression pedal when the document is loaded

These are the properties that can be assigned to a control which is an *alias*:

of patch	the patch that contains the reference control for the linked control
of control	the reference control for the linked control

The "states" section

By defining multiple states, an FCB 1010's footswitch, which normally behaves as a pushbutton, can be transformed into a switch or a selector. Expression pedals always have a single state.

When a footswitch has more than one state, the name of its current state is shown inside the footswitch in the main window.

States contain actions, which are executed in order when the control that contains the state is activated. The number of actions that a state can contain is virtually unlimited.

This section contains the following buttons.

new	creates a new, empty state
duplicate	duplicates the currently selected state
cut	copies the currently selected state into the clipboard and deletes it from the list

copy	copies the currently selected state into the clipboard
paste	creates a new state as a copy of that contained in the clipboard
delete	deletes the currently selected state
clear	deletes all the states
arrows	move the currently selected state up or down in the list

These are the properties that can be assigned to a state of a *multi-state* footswitch:

label	text that identifies the state; it is shown below the footswitch primary label
color	the color with which the name of the state is painted; there is a library of specific colors for the names of states; using the menu that appears when you click on the color, you can choose a color from those available in the library, or add a new one, or modify or delete existing colors

The "actions" section

Actions are FCBBooster's building blocks. An action usually define a particular MIDI message that is sent to a "target", altough a few special non-MIDI actions exist. Each action, according to its kind, has a specific set of properties.

Actions are triggered when one of the ten FCB 1010 footswitches is pressed or released or one of the two expression pedals changes its position. Footswitches and expression pedals use different kind of actions.

This section contains the following buttons.

new	creates a new action; clicking on the button opens a menu to choose the type of the action to create between those which are relevant for the type of the currently selected control
duplicate	duplicates the currently selected action; this is not available when the action is of type "go to patch"
cut	copies the currently selected action into the clipboard and deletes it from the list
copy	copies the currently selected action into the clipboard
paste	creates a new action as a copy of that contained in the clipboard
delete	deletes the currently selected action
clear	deletes all the actions
arrows	move the currently selected action up or down in the list, anticipating or postponing its execution

Common properties for footswitches and expression pedals

Each action has the following properties.

comment	you can use this space for notes about the action
perform if state of footswitch 'x' is	this property can appear from zero to 9 times, depending on how many controls belonging to the same patch have multiple states and have set the "actions see states" property to "yes", not including the control that owns the action; the available choices will include "any", meaning that the action will perform anyway, ignoring the state of the control, plus one item for each state of the control, allowing to specify that the action should perform only when that state is current

Actions for footswitches

The following actions can be assigned to a footswitch.

midi program change

send message to	the target that will receive the message
perform when	indicates whether the action will be executed by pressing or releasing the footswitch
select program	the number (0-127) of the MIDI program that will be selected

midi control change

send message to	the target that will receive the message
change controller	identifies the MIDI controller that will be changed
perform when	indicates whether the action will be executed by pressing or releasing the footswitch
set value	the value (0-127) that will be assigned to the MIDI controller

midi note

send message to	this is the target that will receive the message
note number	the MIDI note (C2-G8) that will be generated
note velocity	the velocity (0-127) of the MIDI note that will be generated

midi sysex

send message to	this is the target that will receive the message
sequence	the sequence of bytes that will be sent to the chosen target; input the bytes as hexadecimal values, separated by a space; do not include the F0 prefix and the F7 suffix; if you need to specify a "Roland checksum", use the "$" character to indicate the starting point of the sequence that must be included in the checksum and the "=" character to indicate the end of the sequence as well as its insertion point
perform when	indicates whether the action will be executed by pressing or releasing the footswitch

midi all sound off

send message to	this is the target that will receive the message
perform when	this indicates whether the action will be executed by pressing or releasing the footswitch

midi all notes off

send message to	this is the target that will receive the message
perform when	this indicates whether the action will be executed by pressing or releasing the footswitch

midi reset controllers

send message to	this is the target that will receive the message
perform when	this indicates whether the action will be executed by pressing or releasing the footswitch

go to patch

select patch	the patch that will be selected by performing the action

mute expression pedal

pedal belongs to patch	the patch that contains the expression pedal to be muted or unmuted by performing the action
pedal is	the expression pedal to be muted or unmuted by performing the action
action	indicates whether the expression pedal will be muted or unmuted

wait

perform when	this indicates whether the action will be executed by pressing or releasing the footswitch
delay time [msec]	the delay amount in milliseconds (0 - 9999)

Actions for expression pedals

The following actions can be assigned to an expression pedal.

midi control change

send message to	the target that will receive the message
change controller	identifies the MIDI controller that will be changed
minimum value	the minimum transmitted value; incoming values will be remapped according to this setting
maximum value	the maximum transmitted value; incoming values will be remapped according to this setting

midi sysex

send message to	this is the target that will receive the message
sequence	the sequence of bytes that will be sent to the chosen target; input the bytes as hexadecimal values, separated by a space; do not include the F0 prefix and the F7 suffix; use the "*" character to indicate where the current value of the expression pedal will be inserted; if you need to specify a "Roland checksum", use the "$" character to indicate the starting point of the sequence that must be included in the checksum and the "=" character to indicate the end of the sequence as well as its insertion point
size of variable data	determines whether the value corresponding to the state of the expression pedal will be sent using 1 or 2 bytes
minimum value	the minimum transmitted value; incoming values will be remapped according to this setting
maximum value	the maximum transmitted value; incoming values will be remapped according to this setting

midi channel pressure

send message to	the target that will receive the message

minimum value	the minimum transmitted value; incoming values will be remapped according to this setting
maximum value	the maximum transmitted value; incoming values will be remapped according to this setting

MIDI Implementation Reference

Functions			Transmitted	Recognized
Basic Channel			1 - 16	1 - 16
Note Number			0 - 127	0 - 127
Channel Voice	Note Off event		O	O
	Note On event		O	O
	Polyphonic Key Pressure		X	X
	Control Change	0 - 127	O	O
	Program Change		O	O
	Channel Pressure		O	X
	Pitch Wheel Change		X	X
Channel Mode	All Sound Off		O	X
	Reset All Controllers		O	X
	Local Control		X	X
	All Notes Off		O	X
System Common	System Exclusive		O	O
	MIDI Time Code		X	X
	Song Position Pointer		X	X
	Song Select		X	X
	Tune Request		X	X
	End of Exclusive		X	X
System Real Time	Clock		X	X
	Start		X	X
	Continue		X	X
	Stop		X	X
	Active Sensing		X	X
	Reset		X	X